

Formularios:

Para el procesamiento de formularios, se requiere disponer de dos instancias; una dedicada a mostrarle al usuario los campos en los cuales ingresará su información, y otra que se encargue de tratar dicha información una vez que sea enviada al servidor.

index.php (text y password)

Vamos por la primera, un formulario se crea con la siguiente etiqueta:

```
<form method="post" action="lib.php" name="frmtxtpass">
</form>
```

Aquí aparecen 3 atributos de **form** que tienen que ver con el envío de la información:

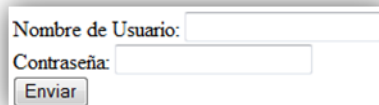
- **Method:** puede ser **POST** (ocultos) o **GET** (por la barra de dirección).
- **action:** aquí se indica la ruta y el nombre de la página que procesará la información.
- **Name:** nombre del formulario, en caso de que existan varios en una misma página.

Para que la información sea enviada necesitamos una entrada de tipo **submit** dentro del formulario

```
<form method="post" action="lib.php" name="frmtxtpass">
  <input type="submit" name="enviar_txtpass" value="Enviar" />
</form>
```

Y para graficar este ejemplo le incluiremos una entrada de tipo **text** y **password** donde le pediremos nombre de usuario y contraseña.

```
<form method="post" action="lib.php" name="frmtxtpass">
  Nombre de Usuario: <input type="text" name="_user" id="" /><br />
  Contraseña: <input type="password" name="_pass" id="" /><br />
  <hr />
  <input type="submit" name="enviar_txtpass" value="Enviar" />
</form>
```



lib.php

Este archivo es quien recibirá la información, lo primero que haremos será averiguar de dónde proviene el llamado, para ello utilizaremos la función **isset(\$variable)** que nos devuelve verdadero en caso de que dicha variable exista o se haya definido anteriormente, y la usaremos para verificar la existencia del botón Enviar cuyo nombre es **enviar_txtpass**,

```
<?php
if (isset($enviar_txtpass)) {}
?>
```

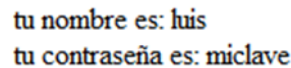
Como verás, los nombres de los campos de un formulario se reciben como nombres de variables

Lo que sigue, es recuperar los campos en los cuales el usuario ingresó sus datos, para ello recurrimos a las variables globales **\$HTTP_POST_VARS[]** o **\$_POST[]** si fueron enviados por **post**, o bien **\$HTTP_GET_VARS[]** o **\$_GET[]** si fueron enviados por **get**

```
<?php
if (isset($enviar_txtpass)) {
  $user=$HTTP_POST_VARS['_user'];
  $pass=$HTTP_POST_VARS['_pass'];
}
?>
```

Notar que en los índices de las variables globales (los que están entre corchetes) aparecen los nombres de los campos del formulario donde se ingresaron los datos. Finalmente mostramos los datos ingresados

```
<?php
if (isset($enviar_txtpass)){
    $user=$HTTP_POST_VARS['_user'];
    $pass=$HTTP_POST_VARS['_pass'];
    echo "tu nombre es: ".$user. "<br />" ;
    echo "tu contraseña es: ".$pass ;
}
?>
```

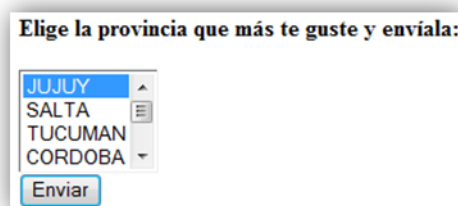


[index.php \(listas de selección\)](#)

Se trata de una lista que permite seleccionar uno de sus elementos. Esta lista es de tipo **SELECT** que en nuestro ejemplo se llama **campolista**, y cada uno de sus elementos es un **option** cuyo **value** es distinto para cada opción y es el que lo identifica. Cuando este formulario se envía, la opción seleccionada toma el **value** de su **option** correspondiente y lo asigna al campo select.

```
<form method="post" action="lib.php" name="frmlistas">
  <b>Elige la provincia que más te guste y enviala:</b><br /><br />
  <select name="campolista" id="" size="4">
    <option value="Jujuy" selected>JUJUY</option>
    <option value="Salta">SALTA</option>
    <option value="Tucumán">TUCUMAN</option>
    <option value="Córdoba">CORDOBA</option>
    <option value="La Rioja">LA RIOJA</option>
    <option value="San Juan">SAN JUAN</option>
    <option value="Mendoza">MENDOZA</option>
  </select><br />
  <input type="submit" name="enviar_lista" value="Enviar" /><br />
</form>
```

El atributo **size="4"** de la etiqueta **Select**, indica que son 4 los elementos de la lista que serán visibles al usuario, el atributo **selected** de la etiqueta **option**, selecciona por defecto al elemento de la lista en el cual se encuentra definido, en este ejemplo selecciona por defecto a JUJUY.



[lib.php](#)

```
<?php
if (isset($enviar_lista)){
    $prov=$HTTP_POST_VARS['campolista'];
    echo "tu provincia preferida es: ".$prov;
}
?>
```

Aquí sólo se ha recuperado la variable que corresponde al campo **select**, el valor de este campo es el **value** del **option** seleccionado de la lista, este sería el resultado de seleccionar Tucumán.

Las leyendas y sus opciones correspondientes, se encuentran relacionadas a fin de permitir seleccionarlas ya sea haciendo clic sobre el texto o sobre dicha opción. Esto se logra mediante el **id** del **radio** y el **for** del **label**. Nota que para cada opción estos atributos tienen el mismo valor.

Por último, el atributo **checked** del **radio**, selecciona por defecto el radio en el cual se encuentra definido, en este caso PHP.

Consulta por uno de los siguientes lenguajes:

HTML CSS JavaScript PHP MySQL

lib.php

lo que haremos en este caso será averiguar cuál de todas las opciones fue la seleccionada, y como son varias utilizaremos la estructura **switch**,

```
<?php
if (isset($enviar_opcion)) {
    $lenguaje=$HTTP_POST_VARS['len'];
    switch ($lenguaje) {
        case "HTML":
            $texto = "El HTML te permitirá crear páginas estáticas simples.";
            break;
        case "CSS":
            $texto = "Con CSS podrás separar del HTML tolo lo relacionado al estilo de tu
                web.<br />";
            $texto = $texto."Incluso podrás hacerlo en un archivo aparte.";
            break;
        case "JS":
            $texto = "JavaScript Se caracteriza por darle dinamismo a la web.<br />";
            $texto .= "Hasta podrás crear efectos especiales...!!!<br />";
            $texto .= "Prueba este enlace: <a href=\"javascript: window.history.back()\">
                regresar</a>";
            break;
        case "PHP":
            $texto = "Demás está hablar de PHP, para darte una idea, esto se creó con PHP.";
            break;
        case "MSQL":
            $texto = "MySQL es el gestor de Bases de Datos más utilizado en Internet<br />";
            $texto .= "podrás por ejemplo; Agregar, Modificar o Eliminar los datos de una";
            $texto .= "Base de Datos.";
            break;
        default:
            $texto = "La idea es que selecciones una opción";
    }
    echo "La opción seleccionada es: ".$lenguaje;
    echo $texto;
}
?>
```

Aquí el resultado al seleccionar CSS:

La opción seleccionada es: CSS
Con CSS podrás separar del HTML tolo lo relacionado al estilo de tu web.
Incluso podrás hacerlo en un archivo aparte.

Index.php (Cuadros de Selección)

Son entradas (**inputs**) de tipo **checkbox**, que permiten seleccionar una o más de las opciones disponibles, en este caso la información se envía como un array[] que contiene todas **las opciones seleccionadas**, por lo cual al atributo **name** de cada **checkbox** se le asigna un mismo array[], en este ejemplo el array se llama **idioma[]**.

```
<form method="post" action="lib.php" name="frmchecks">
  <b>Envíame los Idiomas que dominas con toda confianza:</b><br /><br />
  <input type="checkbox" id="e" name="idioma[]" value="español" checked /> <label
    for="e">Español</label>
  <input type="checkbox" id="i" name="idioma[]" value="italiano" /> <label
    for="i">Italiano</label>
  <input type="checkbox" id="n" name="idioma[]" value="ingles" /> <label
    for="n">Inglés</label>
  <input type="checkbox" id="f" name="idioma[]" value="frances" /> <label
    for="f">Francés</label>
  <br />
  <input type="submit" name="enviar_checks" value="Enviar" />
</form>
```

Igual que antes, el atributo **checked** del **checkbox**, selecciona por defecto las opciones en las cuales se encuentran definidos, en este caso Español.

lib.php

Primero recuperamos el array en otro llamado `$misidiomas` y lo recorremos mediante `foreach`, esta instrucción toma cada elemento de `$misidiomas` y lo pasa a una variable llamada `$lengua` que es la que finalmente mostramos al usuario como si fuese una lista

```
<?php
if (isset($enviar_checks)){
  $misidiomas=$HTTP_POST_VARS['idioma'];
  echo "Los lenguajes que yo domino son:";
  echo "<ul>";
  foreach ($misidiomas as $lengua){
    echo "<li>".$lengua."</li>";
  }
  echo "</ul>";
}
?>
```

Resultado al seleccionar Español e Inglés

Index.php (Áreas de Texto)

Se trata de un área de texto que permite ingresar textos largos, su etiqueta corresponde a `<textarea></textarea>`, los atributos **cols** y **rows** corresponden a las columnas de caracteres y líneas de texto respectivamente, que serán visibles al usuario. Al enviar este formulario, es el value del textarea el que se reciba como información.

```
<form method="post" action="lib.php" name="frmareas">
  <b>Envíame un comentario sobre esta web:</b><br />
  <textarea name="comentario" cols="40" rows="4" id="" >
</textarea>
  <br />
  <input type="submit" name="enviar_textarea" value="Enviar" />
</form>
```

lib.php

El value del textarea lo recuperamos mediante su nombre, en este caso **comentario**.

```
<?php
if (isset($enviar_textarea)){
    $coment=$HTTP_POST_VARS['comentario'];
    echo $coment;
}
?>
```

Index.php (Campos para archivos (File) y campos Ocultos (hidden))

Se trata de entradas de tipo file y hidden respectivamente.

El primero **File**, crea un cuadro de texto y un botón “Examinar” que abre un cuadro de diálogo para seleccionar un archivo, el cual se enviará al servidor.

El segundo **Hidden**, permite definir mediante una variable global **MAX_FILE_SIZE** y su **value**, el tamaño máximo del archivo a subir por este método, en nuestro ejemplo **1000000** bits (1 Mb).

Otro detalle a tener en cuenta y que resulta obligatorio es el atributo enctype en el formulario, que deberá ser multipart/form-data

```
<form enctype="multipart/form-data" method="post" action="lib.php" name="frmarchivo">
  <input type="hidden" name="MAX_FILE_SIZE" value="1000000" id="" />
  Enviar este Archivo: <input name="archivo" type="file" />
  <br /><br />
  <input type="submit" name="enviar_archivo" value="Enviar" />
</form>
```

lib.php

cuando se recibe el formulario anterior, se reciben además una serie de variables que contienen información del archivo recibido, php nos brinda el array **\$_FILES** con las siguientes propiedades, en donde archivo es el nombre del campo con el cual se envió el archivo desde el formulario:

- \$_FILES["archivo"]["name"]**: nombre original del archivo
- \$_FILES["archivo"]["type"]**: tipo mime del archivo, por ejemplo image/gif
- \$_FILES["archivo"]["tmp_name"]**: nombre temporal que utiliza para almacenarse en el servidor.
- \$_FILES["archivo"]["error"]**: n° de error al subir el archivo, o si todo fue bien.
- \$_FILES["archivo"]["size"]**: tamaño en bytes del archivo recibido

En nuestro script, lo primero que haremos será verificar que no se hayan producido errores al recibir el fichero. De estar todo bien, escribiremos todas las variables descritas anteriormente utilizando el ciclo `foreach`, donde la variable `$clave` contiene en identificador de dicha variable, y `$valor` el dato correspondiente.

Finalmente lo subimos con la función: `move_uploaded_file(ubicación_temporal,ubicación_destino)`

```
<?php
if (isset($enviar_archivo)){
    if ($_FILES["archivo"]["error"]==2){
        echo "el archivo es demasiado grande";
    }elseif ($_FILES["archivo"]["error"]==4){
        echo "debe seleccionar un archivo";
    }else{
        foreach($_FILES['archivo'] as $clave=>$valor){
            echo "la clave ".$clave." tiene el valor ".$valor."<br />";
        }
        move_uploaded_file
($_FILES["archivo"]["tmp_name"], "imagen/".$_FILES["archivo"]["name"]);
    }
}
?>
```

Y este es el resultado al subir un archivo llamado casa.txt:

```
la clave name tiene el valor red casa.txt
la clave type tiene el valor text/plain
la clave tmp_name tiene el valor C:\Windows\TEMP\phpBBA1.tmp
la clave error tiene el valor 0
la clave size tiene el valor 89
```

Como verás se guardó el archivo en la carpeta `imagen` que es la que está subrayada, para que la tengas en cuenta, con esto es suficiente para que vayas y veas si lograste subir dicho archivo, pero podríamos hacer más:

```
<?php
if (isset($enviar_archivo)){
    if ($_FILES["archivo"]["error"]==2){
        echo "el archivo es demasiado grande";
    }elseif ($_FILES["archivo"]["error"]==4){
        echo "debe seleccionar un archivo";
    }else{
        foreach($_FILES['archivo'] as $clave=>$valor){
            echo "la clave ".$clave." tiene el valor ".$valor."<br />";
        }
        move_uploaded_file
($_FILES["archivo"]["tmp_name"], "imagen/".$_FILES["archivo"]["name"]);
        echo "<br />";
        echo "Ahora puedes ver el <a
href=\"imagen/".$_FILES['archivo']['name']."\">Archivo que Subiste</a><br /><br />";
    }
}
?>
```

Hemos creado un enlace para ver el archivo que acabamos de subir, mira como nos quedó el resultado ahora...

```

la clave name tiene el valor red casa.txt
la clave type tiene el valor text/plain
la clave tmp_name tiene el valor C:\Windows\TEMP\phpBBA1.tmp
la clave error tiene el valor 0
la clave size tiene el valor 89

```

Ahora puedes ver el [Archivo que Subiste](#)

Y podemos hacer más, por ejemplo, darle al usuario la posibilidad de quitar dicho archivo:

```

<?php
if (isset($enviar_archivo)){
    if ($_FILES["archivo"]["error"]==2){
        echo "el archivo es demasiado grande";
    }elseif ($_FILES["archivo"]["error"]==4){
        echo "debe seleccionar un archivo";
    }else{
        foreach($_FILES['archivo'] as $clave=>$valor){
            echo "la clave ".$clave." tiene el valor ".$valor."<br />";
        }
        move_uploaded_file
($_FILES["archivo"]["tmp_name"],"imagen/".$_FILES["archivo"]["name"]);
        echo "<br />";
        echo "Ahora puedes ver el <a
href='\"imagen/".$_FILES['archivo']['name']."\">Archivo que Subiste</a><br /><br />";
        echo "Puedes quitar este archivo si lo deseas: <br />";
        echo "<form method='\"post\"' action='\"lib.php\">";
        echo "<input type='\"hidden\"' name='\"archivo\"'
value='\"".$_FILES['archivo']['name']."\" />";
        echo "<input type='\"submit\"' name='\"quitararchivo\"' value='\"Borrar Archivo\"
/>";
        echo "</form>";
    }
}
if (isset($quitararchivo)){
    unlink ("imagen/".$_HTTP_POST_VARS['archivo']);
    echo "El archivo ha sido borrado";
}
?>

```

```

la clave name tiene el valor red casa.txt
la clave type tiene el valor text/plain
la clave tmp_name tiene el valor C:\Windows\TEMP\phpBBA1.tmp
la clave error tiene el valor 0
la clave size tiene el valor 89

```

Ahora puedes ver el [Archivo que Subiste](#)

Puedes quitar este archivo si lo deseas:

El último **isset** que aparece en este código es para dar de baja al archivo cuando se presione el botón **Borrar Archivo**. Lo cual se hace con la función `unlink("ruta_de_archivo_a_borrar")`

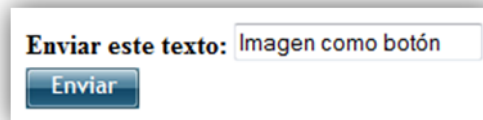
El archivo ha sido borrado

Index.php (Campo de Imagen (image))

El campo **image** envía las coordenadas en donde se hizo clic al momento de enviar el formulario. Por Lo que recurrimos a la implementación de un campo **hidden** para confirmar dicho envío.

Se agregó también un evento **JavaScript** que asigna al campo **hidden** el valor '**botonimag**' el cual será comprobado por nuestro script en **lib.php**

```
<form method="post" action="lib.php" name="frmimagen">
  <b>Enviar este texto:</b>
  <input type="text" name="texto" value="Imagen como botón" id="" />
  <br />
  <input type="hidden" name="btnim" id="" />
  <input type="image" onclick="javascript: frmimagen.btnim.value='botonimag'"
name="btnimagen" src="btn.png" />
</form>
```



Index.php (Campos para archivos (File) y campos Ocultos (hidden))

```
<?php
if (isset($btnim)) {
    echo "Si lees este texto es porque presionaste el botoncito enviar, el cual es
una imagen, como habrás notado.";
}
?>
```

Si lees este texto es por que presionaste el botoncito enviar, el cual es una imagen, como habrás notado.

Quizás más adelante hagamos algo como Formularios Avanzados o Algo así, les parece...???

Saludos...

R-Luis